The importance of temporal information in Bayesian network structure learning

Anthony C. Constantinou[1,2]

1. Bayesian Artificial Intelligence research lab, Risk Information Management research group, School of Electronic Engineering and Computer Science, Queen Mary University of London, London, UK, E1 4NS. E-mail: a.constantinou@qmul.ac.uk

2. The Alan Turing Institute, British Library, 96 Euston Road, London, UK, NW1 2DB.

ABSTRACT - Several algorithms have been proposed towards discovering the graphical structure of Bayesian networks. Most of these algorithms are restricted to observational data and some enable us to incorporate knowledge as constraints in terms of what can and cannot be discovered by an algorithm. A common type of such knowledge involves the temporal order of the variables in the data. For example, knowledge that event B occurs after observing A and hence, the constraint that B cannot cause A. This paper investigates real-world case studies that incorporate interesting properties of objective temporal variable order, and the impact these temporal constraints have on the learnt graph. The results show that most of the learnt graphs are subject to major modifications after incorporating incomplete temporal objective information. Because temporal information is widely viewed as a form of knowledge that is subjective, rather than as a form of data that tends to be objective, it is generally disregarded and reduced to an optional piece of information that only few of the structure learning algorithms may consider. The paper argues that objective temporal information should form part of observational data, to reduce the risk of disregarding such information when available and to encourage its reusability across related studies.

KEYWORDS - causal discovery, causal graphs, directed acyclic graphs, probabilistic graphical models, order-based learning, temporal constraints.

1 INTRODUCTION

A large part of scientific research is driven by interest in discovering causal relationships from data to be used as guides for intervention, to maximise utility of interest and to minimise undesirable risk. Much of this research is based on methods that focus on maximising the predictive accuracy of a target variable X from a set of observed predictors Y. However, the best predictors of X are often not its causes and hence, the motto *association does not imply causation*. While the distinction between association and causation is nowadays better understood, what has changed over the decades is mostly the way the results are stated rather than the way they are produced.

Pearl's and Mackenzie's book (2018) has brought great attention to the importance and need for causal models, like Causal Bayesian Networks (CBNs), as the basis of achieving true AI. Any model that captures cause-and-effect relationships must, by definition, adhere to the temporal order of the variables. For example, an effect at time t can only have causes observed at a time prior to t. The question of how to most effectively develop such models to solve real-world problems is therefore a particularly current concern.

The field of research that appears to have made significant steps towards causal discovery involves the constraint-based algorithms that are typically used to construct Complete Partial Directed Acyclic Graphs (CPDAGs) that can be converted into a BN model. A CPDAG is a

graph that incorporates both directed and undirected edges and represents a Markov equivalence class of Directed Acyclic Graphs (DAGs). Most of the constraint-based algorithms are based on conditional independence tests, amongst others, that generate causal graphs under the assumption that the direction of the edges represents causal or influential relationships between nodes. Undirected edges in a CPDAG represent dependencies whose directionality cannot be determined by observational data. This process is inherited by the *Inductive Causation* (IC) algorithm (Verma and Pearl, 1990). The *Peter and Clark* (PC) algorithm has had a major impact in this area of research due to its simplicity, learning strategies, computational speed, and performance (Glymour and Cooper, 1999; Spirtes et al., 2001).

Alternatives to the constraint-based methods are the score-based algorithms which can be viewed as a traditional machine learning approach. This is because score-based learning involves heuristics that explore the search space of graphs and return the graph that maximises an objective function. Well-established examples include the K2 (Cooper and Herskovits, 1992) and GES (Chickering, 2002) algorithms. Unlike constraint-based methods, score-based algorithms do not make claims about causation. Moreover, hybrid algorithms exist that share characteristics with both the constraint-based and score-based learning, such as the *Max-Min Hill-Climbing* (MMHC) algorithm (Tsamardinos et al., 2006) and the L1-Regularization paths (Schmidt et al., 2007).

While both the constrain-based and score-based algorithms work well in theory (i.e., with synthetic data), for various reasons they are generally less effective when applied to real-world data (Freedman and Humphreys, 1999; Zhang, 2008; Korb and Nicholson, 2011; Koski and Noble, 2012; Dawid et al., 2015). Because of this, BN models are often constructed manually with knowledge, instead of being automatically generated by structure learning algorithms, and this applies to various real-world domains with access to causal knowledge (Fenton and Neil, 2012). As a result, many of the algorithms are defined and developed in ways that enable us to incorporate knowledge about what can and cannot be discovered by the algorithm with reference to the input data. Perhaps the most common type of knowledge involves the temporal order of variables, such as specifying that event B occurs after observing A and hence, B cannot cause A.

The paper is structured as follows: Section 2 provides a formal introduction to BN structure learning with temporal constraints, Section 3 presents the methodology used to perform the experiments, Section 4 describes the experiments and presents the results, and Section 5 discusses the results and provides the concluding remarks.

2 TEMPORAL CONSTRAINTS IN BAYESIAN NETWORK STRUCTURE LEARNING

This section focuses on the standard score-based and constraint-based classes of learning to describe the process of incorporating temporal constraints into the structure learning process.

2.1 Temporal constraints in score-based learning

Cooper and Herskovits' K2 algorithm (1991) represents the first important attempt at learning the graphical structure of BNs. K2 is a score-based algorithm that uses an objective function to score graphs. Specifically, it searches for graph G in data D and a discrete variable set Z that maximises (Cooper and Herskovits, 1991)

$$\max_{G}[P(G,D)] = c\prod_{i=1}^{n}\max_{\pi_i}\left[\prod_{j=1}^{q_i}\frac{(r_i-1)!}{\left(N_{i_j}+r_i-1\right)!}\prod_{k=1}^{r_i}a_{i_{j_k}}!\right] \qquad (1)$$

where n is the number of variables in D, c is a constant ignorant prior probability for each G, π_i is the Candidate Parent Set (CPS) of variable $x_i \in Z$, r_i is the number of unique instantiations of x_i, q_i is the number of unique instantiations of π_i, $a_{i_{j_k}}$ is the number of cases in D in which variable x_i is instantiated as v_{i_k} and CPS π_i is instantiated as ϕ_{i_j}, and

$$N_{i_j} = \sum_{k=1}^{r_i} a_{i_{j_k}} \qquad (2)$$

K2 is also an order-based algorithm which assumes that complete information about the temporal ordering of the variables is given. Full prior information of the ordering eliminates the need to assess the orientation of edges. This is because the temporal information is imposed as a directionality constraint in the search space of graphs. In equation (1), this constraint translates into pruning of the CPS π_i for each variable $x_i \in Z$. For example, if variable x_i precedes variable x_j in the ordering, then $x_j \rightarrow x_i$ would violate the ordering. To ensure such a violation does not occur, the CPS π_i of x_i would need to be pruned by removing all the combinations of parents that include x_j.

Complete information of the ordering represents a very strong, and often unrealistic, assumption that greatly reduces the search space of possible graphs. Specifically, full knowledge of the temporal ordering reduces the search space from super-exponential into $2^{\frac{n^2-n}{2}}$, which remains exponential in n.

2.2 Temporal constraints in constraint-based learning

The PC algorithm (Spirtes et al., 2001) is one of the oldest and most important constraint-based algorithms. Unlike score-based learning that relies on a search-and-score process, constraint-based learning involves constructing a graph that is consistent with the results obtained over a series of conditional independence tests. The PC algorithm is based on the following six main steps (Spirtes et al., 2001):

i. Forms a fully connected undirected graph G where each variable x_i is linked to all other variables that belong in Z.

ii. Eliminates edges in G with a marginal dependency score lower than a given significance threshold a (the threshold is usually set to $a = 0.01$ or $a = 0.05$).

iii. Performs conditional independence tests for each remaining edge $x_i - x_j$ in G, where $x_i - x_j$ is removed if x_i and x_j are found to be independent conditional on a third variable x_k that is connected to either x_i or x_j; i..e. if $x_i \perp\!\!\!\perp x_j | x_k$.

iv. Performs conditional independence tests for each remaining edge $x_i - x_j$ in G, where $x_i - x_j$ is removed if x_i and x_j are found to be independent conditional on a pair of variables $\{x_k,\ x_l\}$ with edges both connected to x_i or both connected to x_j; i.e., if $x_i \perp\!\!\!\perp x_j | \{x_k,\ x_l\}$.

v. For each triple of variables connected as $x_i - x_j - x_k$, it orientates the triple as a v-

structure (also known as the causal class of common-effect) $x_i \rightarrow x_j \leftarrow x_k$ if x_j did not appear in the conditioning set from which A and B had their edge eliminated.

vi. For each triple of variables connected as $x_i \rightarrow x_j - x_k$, it orientates edge $x_j - x_k$ as $x_j \rightarrow x_k$.

In a constraint-based learning process similar to PC, temporal constraints would influence learning steps v and vi. Specifically, partial temporal information would determine some of the edges preserved at the end of step iv, thereby pruning any tests needed to determine the orientation of those edges. In the case of complete temporal information, the orientation of the edges would be determined exclusively by the temporal constraints. This would make steps v and vi redundant, and the output graph a DAG (rather than a CPDAG).

Since constraint-based learning focuses on the exploration of local structures in sets of triples, as opposed to iterating over global structures as in score-based learning, it is generally considered to have less computational complexity than score-based learning. As a result, temporal constraints are likely to have less impact on the computational complexity of a constraint-based algorithm compared to the impact they may have on the computational complexity of a score-based algorithm.

3 METHODOLOGY

We often have partial, and rarely complete, information about the temporal order of the variables in the data. The methodology is driven by interest in assessing a) the ability of some well-established structure learning algorithms in terms of discovering graphs that satisfy known undisputed temporal facts, and b) the benefit of incorporating such temporal information as constraints into the structure learning process of these algorithms. The subsections that follow provide details about the case studies, the data, and the structure learning algorithms considered.

3.1 Data and Case Studies

Three case studies are considered that come from applications of BN modelling in different real-world domains. All three case studies incorporate interesting properties of temporal variable order suitable for the purposes of this paper (discussed in Section 4). The properties of the datasets associated with each case study are depicted in Table 1.

Table 1. The properties of the datasets for each case study.

Data details	Football performance	Forensic psychiatry	Property market
# of variables	7	56	27
Sample size	380	953	1,000
Variable type	Continuous	Discrete	Discrete
Missing values	No	Yes	No

The first case study, which represents the simplest of the three, is based on football (soccer) team performance statistics taken from the English Premier League season 2017/18. In football, teams aim to gain possession (P) of the ball so that they can create shots (S) on target (T) to score a goal (G) when the keeper fails to save the shot. While there are various other fac-

tors that influence the outcome of the variables defined here, there is a transparent and objective temporal order, from P to G, as illustrated in Figure 1.

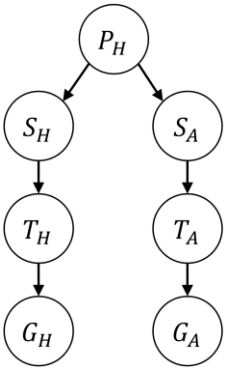


Fig 1. The assumed 'true' BN model of the football performance case study, where P is possession, S is shots created, T is shots on target, and G is goals scored, for both home (H) and away (A) teams.

A sample of the first 10 rows of the dataset is provided in Table 2. Note that for variable *Possession* we only need to know the possession of the home team P_H, since the possession of the away team P_A is $P_A = 1 - P_H$. The data for variables S, T and G are collected from football-data.com, and the data for variable P from whoscored.com.

Table 2. The first 10 rows, out of 380, of the football performance dataset, where P is possession, S is shots, T is shots on target, and G is goals scored, for both home (H) and away (A) teams.

P_H	S_H	S_A	T_H	T_A	G_H	G_A
0.7	27	6	10	3	4	3
0.22	6	14	2	4	0	2
0.62	19	10	6	5	2	3
0.57	14	8	4	6	0	3
0.62	9	9	4	1	1	0
0.6	29	4	2	0	0	0
0.46	9	14	4	5	3	3
0.29	16	9	6	2	1	0
0.55	22	9	6	1	4	0
0.27	6	18	3	6	0	2

The second and relatively complex case study is based on the forensic psychiatry data taken from (Constantinou et al., 2015). These data capture information about released prisoners with serious history of violence and mental health problems in the UK. The 56 variables that make up the data are listed in Table 3. Some of the variables are based on transparent and objective temporal observations and associate with events occurred before serving prison sentence, during prison, and after release from prison. Observations related to events that occurred before, during, and after serving prison sentence are indicated as temporal tiers t1, t2 and t3 respectively, where t1 precedes t2 and t2 precedes t3. Observations not necessarily belonging to a particular temporal tier are indicated with 'n/a'.

Table 3. The data variables of the forensic psychiatry case study. Observations that associate with events occurred before, during, and after prison sentence are indicated with the respective temporal tiers of t1, t2, and t3. Observations not necessarily belonging to a particular temporal tier are indicated with 'n/a'.

Variable name	Acronym	Temporal tier	Variable name	Acronym	Temporal tier
Level of education	E	t1	Cannabis dependence	CND	n/a
Prior violent convictions	PVC	t1	Cocaine dependence	CCD	n/a
Prior acquisitive crime	PAC	t1	Ecstasy dependence	ED	n/a
Abuse or neglect as child	AB	t1	Alcohol dependence	AD	n/a
Cannabis use before prison	CNBP	t1	Gang member	GM	n/a
Cocaine use before prison	CCBP	t1	Violent thoughts	VT	n/a
Ecstasy use before prison	EBP	t1	Negative attitude	NA	n/a
Crim. family background	CB	t1	Criminal attitude	CA	n/a
Cannabis during prison	CNDP	t2	Criminal network	CN	n/a
Cocaine use during prison	CCDP	t2	Victimisation	V	n/a
Ecstasy use during prison	EDP	t2	Living circumstances	LC	n/a
Cannabis use after release	CNR	t3	Social withdraw	SW	n/a
Cocaine use after release	CCR	t3	Employment or training	ET	n/a
Ecstasy use after release	ER	t3	Ability to cope	AC	n/a
Hazard. drinking after release	HD	t3	Stress	S	n/a
Time since release	TR	t3	Domestic stability	DS	n/a
Compliance supervision	CS	t3	Financial difficulties	FD	n/a
Age	AGE	n/a	Problematic life events	PLE	n/a
Gender	G	n/a	Anxiety	AX	n/a
Intelligence/IQ	I	n/a	Depression	D	n/a
Anger problems	AN	n/a	Mental illness symptoms	MI	n/a
Impulsivity problems	IM	n/a	Strange experiences	SE	n/a
Borderline person. Disorder	BPD	n/a	Thought insertion	TI	n/a
Antisocial person. Disorder	ASPD	n/a	Hallucinations	H	n/a
Psychopathy checklist score	PCLRS	n/a	Paranoid delusions	PD	n/a
Psychopathy score (f1)	PCLRF1	n/a	Failed to attend therapy	FT	n/a
Psychopathy score (f2)	PCLRF2	n/a	Responsiveness to treatment	RT	n/a
Psychopathy score (f3)	PCLRF3	n/a	Violence	VI	n/a

The third case study is based on the property market BN model presented in (Constantinou and Fenton, 2017). This BN model was used to assess the impact of property investment tax reforms introduced in 2015 by the British government. The 27 variables that make up the model are listed in Table 4 and ordered by temporal tier. The temporal order of the variables is based on clearly defined rules and regulating protocols that associate with the UK's Buy-To-Let property sector. Specifically, variables at temporal tier t1 involve features that associate with the purchase of the property, at t2 they involve features that associate with rental income and expenses for the year following the purchase of the property, at t3 they involve features that associate with tax expenses and net profit given t2, at t4 they involve features that associate with the future growth in property value, and at t5 they involve features that associate with the future growth in rental income and associated expenses.

It is important to highlight that while all the variables belong to a specified temporal tier, this information does not constitute complete temporal information. This is because edges between variables that fall within the same tier are not subject to temporal constraints. Moreover, unlike the previous two case studies which involve real data, this third case study involves synthetic data generated directly from the conditional distributions of the BN model. Since synthetic experiments tend to overestimate real-world performance, this third case study investigates whether the conclusions obtained from synthetic data are consistent with those obtained from real data.

Table 4. The variables that make up the property market dataset, and the temporal tier for each variable.

Variable name	Acronym	Temporal tier	Variable name	Acronym	Temporal tier
Property purchase value	[PPV]	t1	Rental income gross profit	[RGP]	t3
Stamp duty tax band	[SDTB]	t1	Rental income gross yield	[RGY]	t3
Stamp duty tax	[SDT]	t1	Rental income profit before interest	[RIBI]	t3
Borrowing	[B]	t1	Net profit	[NP]	t3
Loan-To-Value	[LTV]	t1	Income tax	[IT]	t3
Rental income	[RI]	t2	Interest tax relief	[ITR]	t3
Rental income loss	[RIL]	t2	Property value t+1	[PVT1]	t4
Actual rental income	[ARI]	t2	Capital growth	[CGR]	t4
Property expenses	[PE]	t2	Capital gains	[CGA]	t4
Property management expenses	[PME]	t2	Rental Income t+1	[RIT1]	t5
Other property expenses	[OPE]	t2	Rental growth	[RG]	t5
Interest	[I]	t2	Other property expenses t+1	[OET1]	t5
Interest rate	[IR]	t2	Property expenses growth	[PEG]	t5
Other interest expenses	[OIE]	t2			

3.2 Structure learning algorithms

Since many of the experiments involve incorporating partial temporal information as constraints into the structure learning process, the selection of the algorithms is restricted to those that accept such partial constraints. Moreover, the algorithms would also need to work with both continuous and discrete data, as well as with datasets that incorporate missing values. The TETRAD freeware provides access to six well-established structure learning algorithms, spanning all three classes of learning (i.e., constraint-based, score-based and hybrid), that satisfy these requirements. While each algorithm comes with a set of parameters that could be manipulated by the user, such as the level of significance α described in subsection 2.2, we shall investigate the algorithms with their parameter defaults as implemented in TETRAD v6.5.3. Note that these parameters are not intended for tuning on a given dataset; they represent optional thresholds that can be subjectively manipulated to produce denser, or less dense, graph. The six algorithms considered and are briefly discussed below.

Perhaps the most well-known constraint-based algorithm is the PC algorithm previously described in subsection 2.2. Here we consider the modern version of the PC algorithm, called PC-Stable, that solves PC's order dependency issue determined by the order of the variables as they appear in the data (Colombo and Maathuis, 2014). The PC-Stable generates CPDAGs from a set of d-separation equivalence classes of DAGs under the assumption that no latent common causes exist (Spirtes and Glymour, 1991). A variant of the PC algorithm is also considered, called *Fast Adjacency Search* (FAS). This algorithm only performs the adjacency search of the PC algorithm and hence, it returns the skeleton of PC (Spirtes et al., 2001).

The *Fast Causal Inference* (FCI) algorithm is a constraint-based algorithm that, unlike other PC variants, accounts for the possibility of latent variables. Similar to the PC algorithm, it performs a series of conditional independence tests to determine which edges to eliminate, starting from a fully connected undirected network. It then proceeds to the orientation phase that uses the stored conditioning sets that had led to the removal of adjacencies at the previous step, to orientate as many of the preserved edges as possible (Spirtes et al., 2001; TETRAD, 2017). The *Really Fast Causal Inference* (RFCI) algorithm is also considered, which is a variant of the FCI algorithm that decreases runtime by performing fewer conditional independence tests that are conditioned on a smaller set of variables, at the expense of minor changes to the output graph (Colombo et al., 2012).

The fifth algorithm considered is the *Fast Greedy Equivalent Search* (FGES) which represents an optimised version of the *Greedy Equivalence Search* (GES) algorithm that was initially developed by Meek (1997) and later further developed by Chickering (2002). Unlike the four constraint-based algorithms discussed above, the FGES is a score-based algorithm that returns the graph that maximises the Bayesian score via greedy search.

Lastly, the *Greedy Fast Causal Inference* (GFCI) algorithm is considered which combines the FGES and FCI algorithms discussed above, thereby forming a hybrid structure learning process. This combination aims to improve both the accuracy as well as the efficiency by supplementing the initial set on nonadjacencies of FGES with a series of conditional independence tests of FCI to eliminate further adjacencies (Spirtes et al., 2001; Ogarrio et al., 2016).

4 EXPERIMENTS AND RESULTS

The results are presented per case study in the subsections that follow. A set of accuracy metrics is also used to quantify the accuracy of the learnt graphs with respect to the ground truth graphs. These metrics are based on the confusion matrix parameters where True Positives (TP) is the number of true edges discovered in the generated graph, False Positives (FP) is the number of false edges discovered in the generated graph, True Negatives (TN) is the number of true direct independencies in the generated graph, and False Negatives (FN) is the number of false direct independencies in the generated graph. The scoring metrics considered come from the relevant literature. These are:

i. the Precision (Pr) and Recall (Re) defined as $Pr = \frac{TP}{TP+FP}$ and $Re = \frac{TP}{TP+FN}$ respectively,

ii. the F1 score defined as

$$F1 = 2\frac{Pr.Re}{Pr + Re}$$

iii. the SHD score (Tsamardinos et al., 2006) defined as $SHD = FN + FP$, and

iv. the BSF score (Constantinou, 2019) defined as

$$BSF = \left(\frac{TP}{a} + \frac{TN}{i} - \frac{FP}{i} - \frac{FN}{a}\right)\Big/_2$$

where a is the number of edges in the true graph and i is the number of direct independencies in the true graph defined as

$$i = {|Z|(|Z| - 1)}/_2$$

where Z is the variable set as defined in Section 2, and $|Z|$ is the size of variable set Z.

In this study, the above metrics are used to measure the impact of temporal constraints, rather than to measure the accuracy of the different algorithms considered.

4.1 Case study 1: Football team performance

Table 5 presents the graphs generated by each of the six algorithms over the different temporal constraints. The position of the nodes depicted in each of the graphs in Table 1 is based on the position of the nodes as shown in Fig 1. The first column presents the graphs generated without any temporal constraints, whereas the remaining columns progressively increase the amount of temporal information provided as temporal constraints into the structure learning process of each algorithm. Specifically, the constraint $P \rightarrow S, T, G$ involves partial ordering of the nodes specifying that P occurs first in the temporal space, the constraint $P \rightarrow S \rightarrow T, G$ involves partial ordering where S occurs after observing P and $\{T, G\}$ occur after observing P and S, whereas the constraint $P \rightarrow S \rightarrow T \rightarrow G$ involves complete ordering of the nodes (assuming the variables S, T, and G are duplicates; one for each team).

Without temporal constraints, the results show that the four constraint-based algorithms PC-Stable, FAS, FCI, and RFCI, are in agreement in determining the edges, although with some disagreements in the orientation of some of those edges. On the other hand, the score-based FGES and hybrid-based GFCI have produced a different set of edges that is in agreement between the two of them, as well as in agreement with the true graph shown in Fig 1. However, and excluding FAS which returns a skeleton, the RFCI, GFCI and FGES failed to orientate any of the edges.

The partial ordering $P \rightarrow S, T, G$ has led to improvements for most of the algorithms, and these are coloured in green. Interestingly, this single piece of temporal information enabled FGES and GFCI to correctly direct all the previous undirected edges and to successfully generate the true graph. The RFCI is the only algorithm that demonstrated both corrections as well as some incorrect revisions which are coloured in red. The extended partial ordering $P \rightarrow S \rightarrow T, G$ and complete ordering $P \rightarrow S \rightarrow T \rightarrow G$ have led to further graphical revisions that do not include any incorrect revisions. Interestingly, while the temporal constraints assisted the algorithms in determining the correct orientation of the edges, the constraints did not lead to any edge additions nor deletions. As a result, only the FGES and GFCI algorithms managed to recover the true graph whose initial set of edges match the edges in the true graph.

Table 5. The graphs generated by each of the six algorithms given the football performance dataset, and under the different temporal constraints. Graphical revisions that are improvements are coloured green, whereas incorrect revisions are coloured red. The position of the nodes is based on Fig 1.

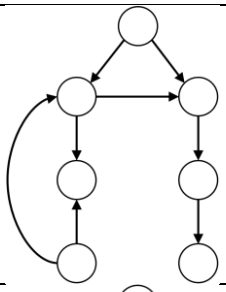
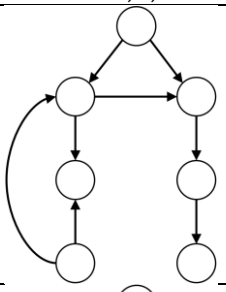
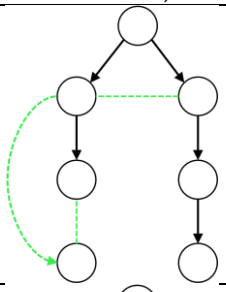
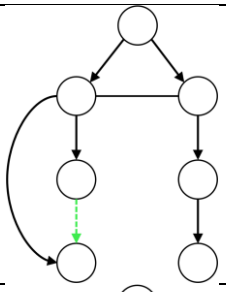
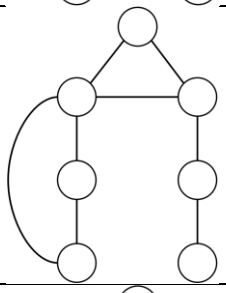
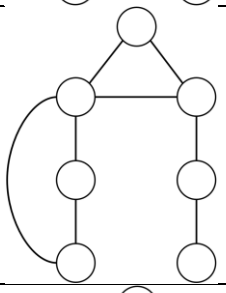
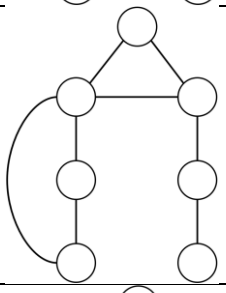
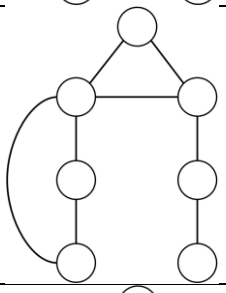
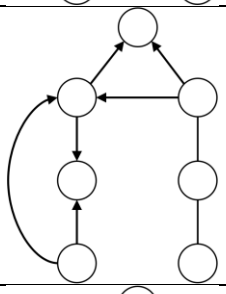
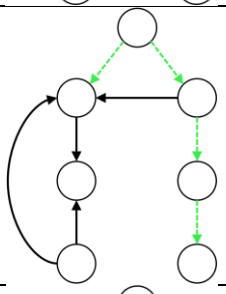
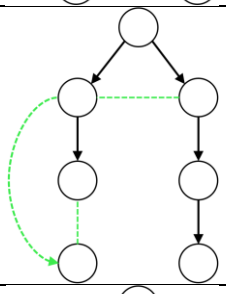
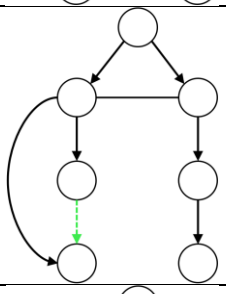
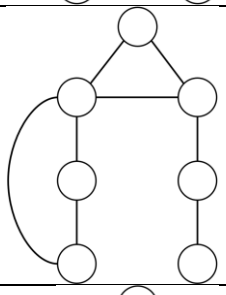
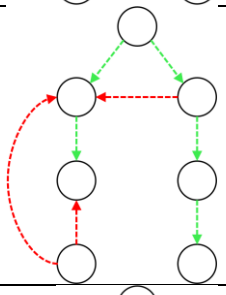
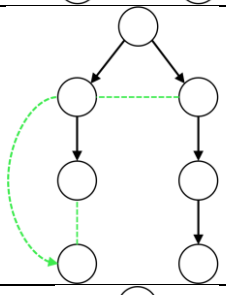
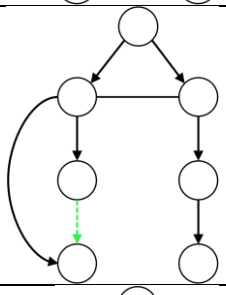
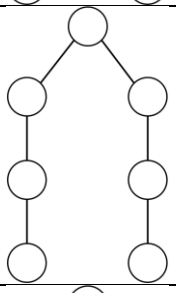
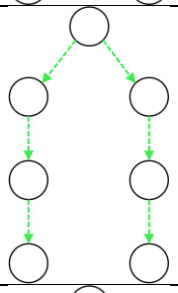
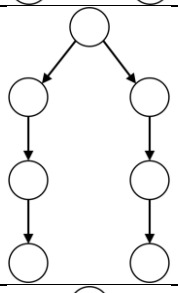
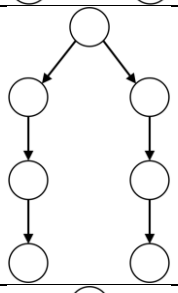
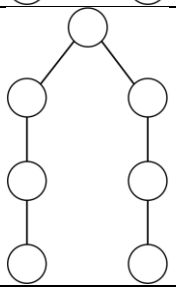
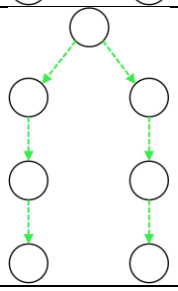
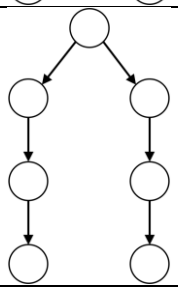
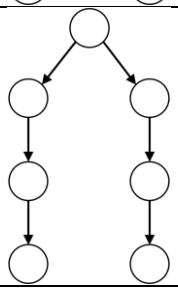
	No temporal constraints	Temporal constraints: $P \rightarrow S, T, G$	Temporal constraints: $P \rightarrow S \rightarrow T, G$	Temporal constraints: $P \rightarrow S \rightarrow T \rightarrow G$
PC STABLE				
FAS				
FCI				
RFCI				
FGES				
GFCI				

Table 6 provides edge statistics for each of the graphs depicted in Table 5 and with reference to each level of temporal constraints. The edge statistics are reported with reference to the true graph shown in Fig 1. Specifically,

→ is the number of directed edges in the learnt graph that are matched in the true graph (also equivalent to TP),
− is the number of edges in the true graph that are undirected in the learnt graph,
← is the number of edges in the true graph that are reversed in the learnt graph,
− is the number of undirected edges in the learnt graph that do not exist in the true graph,
→ is the number of directed edges in the learnt graph that do not exist in the true graph (also equivalent to FP).

Table 6. Edge statistics for each algorithm and over each level of temporal constraints. The edge statistics are reported with reference to the true graph shown in Figure 1.

	No temporal constraints					**Temporal constraints** $P \rightarrow S, T, G$					**Temporal constraints** $P \rightarrow S \rightarrow T, G$					**Temporal constraints** $P \rightarrow S \rightarrow T \rightarrow G$				
Algorithm	→	−	←	−	→	→	−	←	−	→	→	−	←	−	→	→	−	←	−	→
PC-Stable	5	0	1	0	2	5	0	1	0	2	5	1	0	1	1	6	0	0	1	1
FAS	0	6	0	2	0	0	6	0	2	0	0	6	0	2	0	0	6	0	2	0
FCI	1	2	3	0	2	5	0	1	0	2	5	1	0	1	1	6	0	0	1	1
RFCI	0	6	0	2	0	5	0	1	0	2	5	1	0	1	1	6	0	0	1	1
FGES	0	6	0	0	0	6	0	0	0	0	6	0	0	0	0	6	0	0	0	0
GFCI	0	6	0	0	0	6	0	0	0	0	6	0	0	0	0	6	0	0	0	0
Total	6	26	4	4	4	27	6	3	2	6	27	9	0	5	3	30	6	0	5	3

The edge statistics in Table 6 show that, without temporal constraints, only one out of the five algorithms (excluding the adjacency algorithm FAS) managed to discover most of the true arcs (the PC-Stable), whereas the four remaining algorithms failed to discover any of the true arcs; although they did discover most of the true dependencies. As previously mentioned, the single piece of temporal information $P \rightarrow S, T, G$ enabled the algorithms to recover most of the true graph, and two of the algorithms, FGES and GFCI, to fully recover the true graph.

The graphs in Fig 2 illustrate how the graphical revisions translate in terms of accuracy, as determined by each of the metrics. Note that, in contrast to the metrics on the primary axis, a lower SHD score (i.e., error) on the secondary axis indicates a better performance. The results show that even incomplete temporal information would often increase the accuracy scores from less than 0.5 to 1 (or close to 1). The SHD error decreases at a similar rate over the incremental temporal constraints. Specifically, and excluding the adjacency search FAS, the accuracy scores (BSF, F1, Pr, Re) have improved on average by 79% and the SHD error has decreased on average by 67.1%, when comparing the graphs learnt without temporal information to the graphs learnt with complete temporal information. Overall, the scores generated by the metrics are consistent with the graphical revisions illustrated in Table 5 and the edge statistics in Table 6.

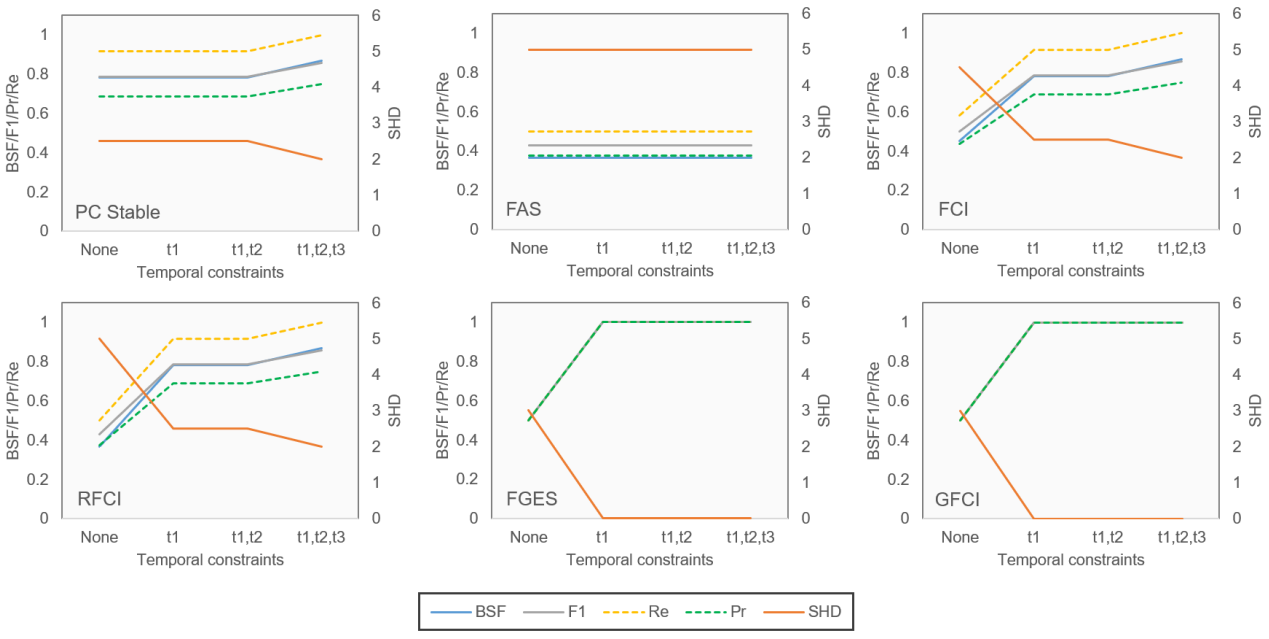


Fig 2. The accuracy scores derived from the scoring metrics Pr, Re, F1 and BSF (Primary axis) and SHD (Secondary axis) for the football team performance (first) case study. Each graph represents an algorithm and illustrates the metric scores change given the temporal constraints.

4.2 Case study 2: Forensic psychiatry

While in the first case study we had complete information about the temporal order of the variables in the data, we have incomplete temporal information in this second case study. Specifically, we know three of the 56 possible temporal orderings, with 17 out of the 56 variables assigned a temporal tier, as shown in Table 3. Further, and contrary to the first case study, the data now consist of discrete variables and incorporate missing values.

Figs 3 and 4 present the graphs generated by each of the algorithms given the temporal constraints specified in Table 3. Note that contrary to the dashed coloured edges in Table 6 which indicate correct and incorrect graphical revisions, the coloured solid edges in Figs 3 and 4 indicate different *types* of graphical revisions. Specifically, additional edges resulting from the temporal constraints are shown in blue colour, reoriented edges (including undirected) are shown in green colour, and edges deleted are shown in red colour.

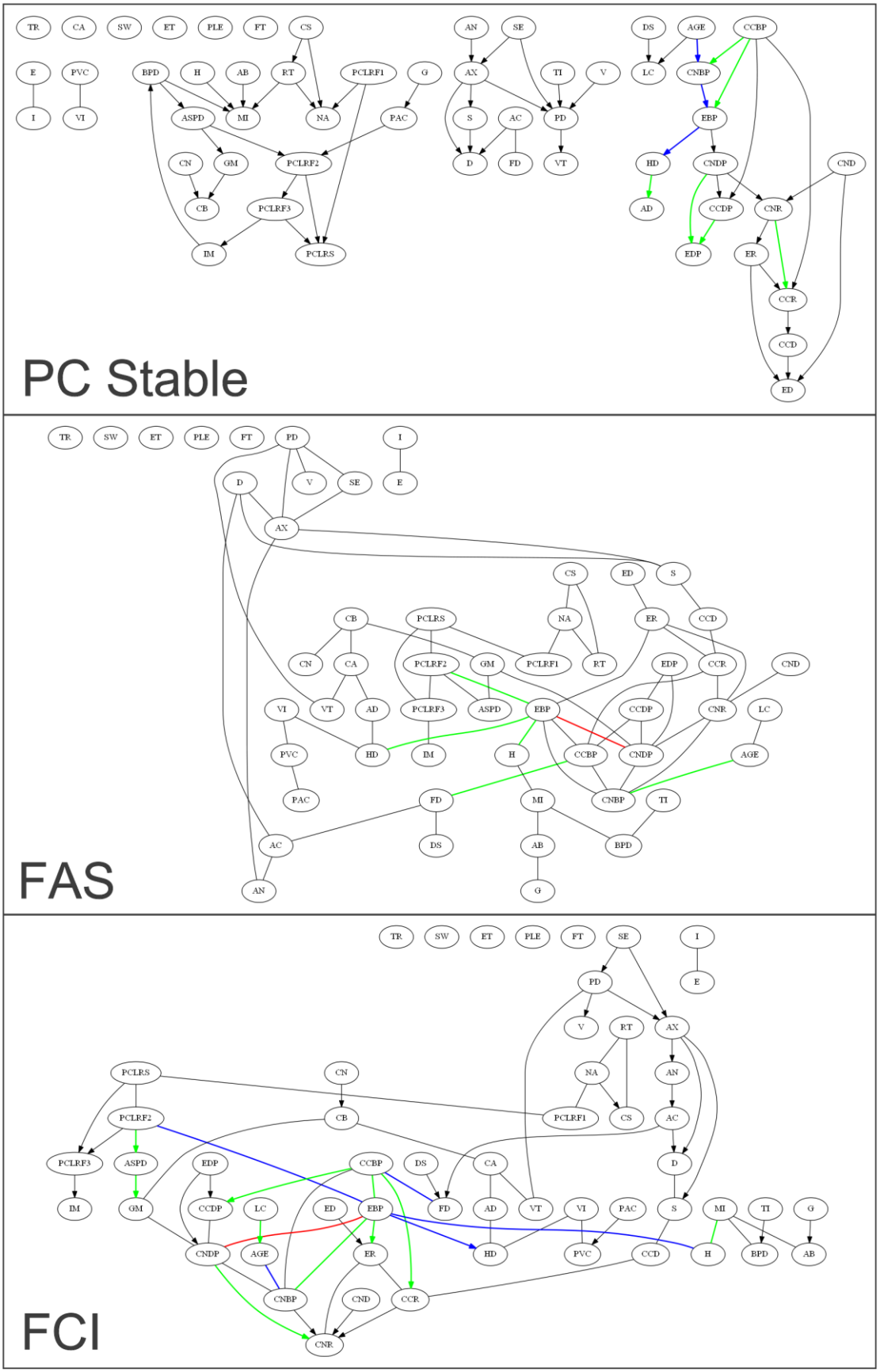


Fig 3. The graphs generated by PC-Stable, FAS, and FCI algorithms, based on the forensic psychiatry case study and the temporal constraints specified in Table 3. New edges resulting from the temporal constraints are shown in blue colour, reoriented edges (including undirected) are shown in green colour, and edges deleted are shown in red colour.

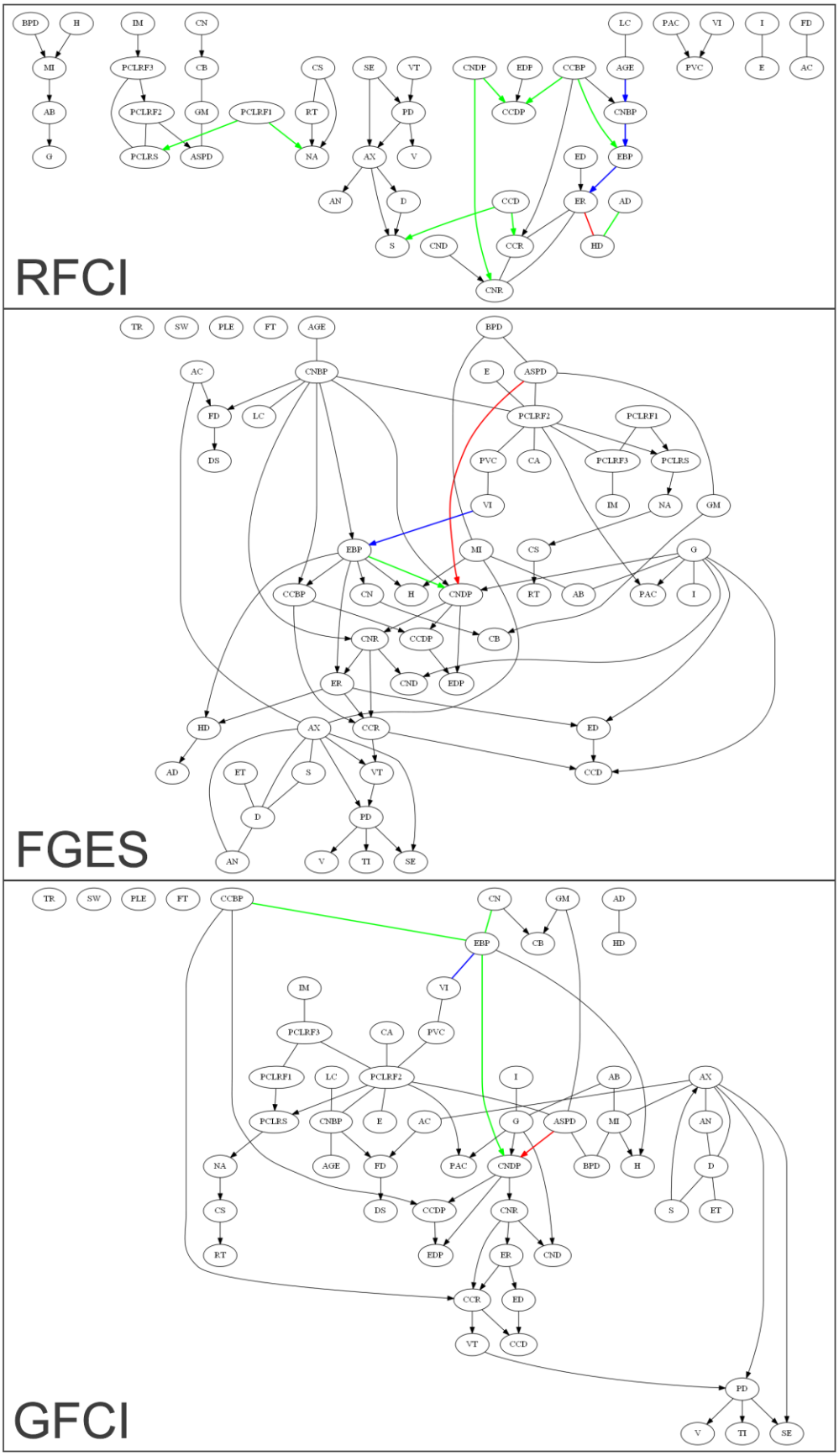


Fig 4. The graphs generated by RFCI, FGES, and GFCI algorithms, based on the forensic psychiatry case study and the temporal constraints specified in Table 3. New edges resulting from the temporal constraints are shown in blue colour, reoriented edges (including undirected) are shown in green colour, and edges deleted are shown in red colour.

Unlike the first case study where the temporal constraints led to only edge reorientations, Figs 3 and 4 show that the graphical revisions in this second case study include edge additions and edge deletions, despite providing only partial information about the temporal order of the variables. However, according to Table 7 which compares the metric scores of the graphs learnt without constraints to the scores of the graphs learnt with constraints, the temporal constraints in this second case study have not led to the same level of improvement as in the first case study.

Overall, the temporal information in Table 3 has modestly improved the scores of constraint-based algorithms PC-Stable, FCI, and FGES, with no changes in the accuracy scores of the score-based FGES and hybrid-based GFCI algorithms despite the minor revisions illustrated in Fig 4. Moreover, the adjacency search FAS was negatively affected by the temporal constraints, suggesting that the improvements observed in the other algorithms are due to modifications in the directionality of the edges (from which FAS cannot benefit since it produces a skeleton graph), rather than edge additions and deletions. Specifically, and excluding the adjacency search FAS, the accuracy scores (BSF, F1, Pr, Re) have improved on average by 7.4%, whereas the SHD error has increased by an average of 0.5%. The conflicting conclusion between the SHD error and the other metrics is an observation documented in (Constantinou, 2019), explained by the fact that the SHD score represents classic classification accuracy whereas the other metrics are designed to offer a more balanced score.

Table 7. The accuracy scores produced by each of the metrics and for each of the algorithms, with and without the temporal constraints specified in Table 3. Green coloured scores improvements and red coloured scores indicate that the temporal constraints have led to an inferior score.

Algorithm	Temporal constraints	Pr	Re	F1	BSF	SHD
PC-Stable	No	0.164	0.094	0.119	0.065	129
PC-Stable	Yes	**0.164**	**0.099**	**0.123**	**0.068**	**131.5**
FAS	No	0.123	0.078	0.096	0.046	134.5
FAS	Yes	**0.115**	**0.078**	**0.093**	**0.043**	**138.5**
FCI	No	0.131	0.083	0.102	0.051	134
FCI	Yes	**0.154**	**0.104**	**0.124**	**0.07**	**136**
RFCI	No	0.16	0.078	0.105	0.053	124.5
RFCI	Yes	**0.194**	**0.099**	**0.131**	**0.073**	**123.5**
FGES	No	0.145	0.115	0.128	0.072	147
FGES	Yes	**0.145**	**0.115**	**0.128**	**0.072**	**147**
GFCI	No	0.114	0.078	0.093	0.04	143.5
GFCI	Yes	**0.114**	**0.078**	**0.093**	**0.04**	**143.5**

4.3. Case study 3: Property market

As discussed in Section 3.1, the third case study differs from the first two case studies in that the structure learning process is based on synthetic, rather than real, data that has been sampled directly from the conditional distributions of the property market BN model. Moreover, the data are discrete and complete. The temporal information involves five temporal tiers, out of a possible of 27 tiers, with all the 27 variables assigned to a temporal tier as shown in Table 4.

Figs 5 and 6 present the graphs generated by each of the algorithms over the different levels of temporal constraints. As in subsection 4.2, new edges resulting from the temporal constraints are shown in blue colour, reoriented edges (including undirected) are shown in green colour, and edges deleted are shown in red colour. The number of revisions observed in this third case study is noticeably higher compared to the number of revisions observed in the second case study, despite the size of the network being approximately half in this case study; i.e.,

27 variables in this case study versus 56 variables in the previous case study. The difference in the number of revisions can be explained by the difference in the number of temporal constraints. Specifically, in the second case study just 17 out of the 56 variables were assigned to one of the possible three temporal tiers, whereas in this case study all the 27 variables are assigned to one of the five temporal tiers.

Since in this case study all the variables associate with a temporal tier, we can illustrate how each additional temporal tier influences the previously learnt graph, as in the first case study. The graphs in Fig 5 illustrate this effect, as determined by each of the metrics. Similar to the first case study and contrary to the second case study, the results from PC-Stable, RFCI and GFCI suggest that partial temporal information, and in this case a single tier (out of possible 27 tiers) of temporal information that includes five out of the 27 variables (i.e., t1 as defined in Table 4), will often lead to important corrections in the learnt graph. Conversely, the graphs learnt by FAS, FCI and FGES demonstrate improvements only after incorporating the first three temporal tiers (i.e., t1, t2, t3) as constraints. Overall, the results are rather consistent across algorithms and show that (excluding the adjacency search FAS) the accuracy scores (BSF, F1, Pr, Re) have improved on average by 45.6% and the SHD error has decreased on average by 43.3%, when comparing the graphs without temporal information to the graphs with the temporal constraints specified in Table 4.

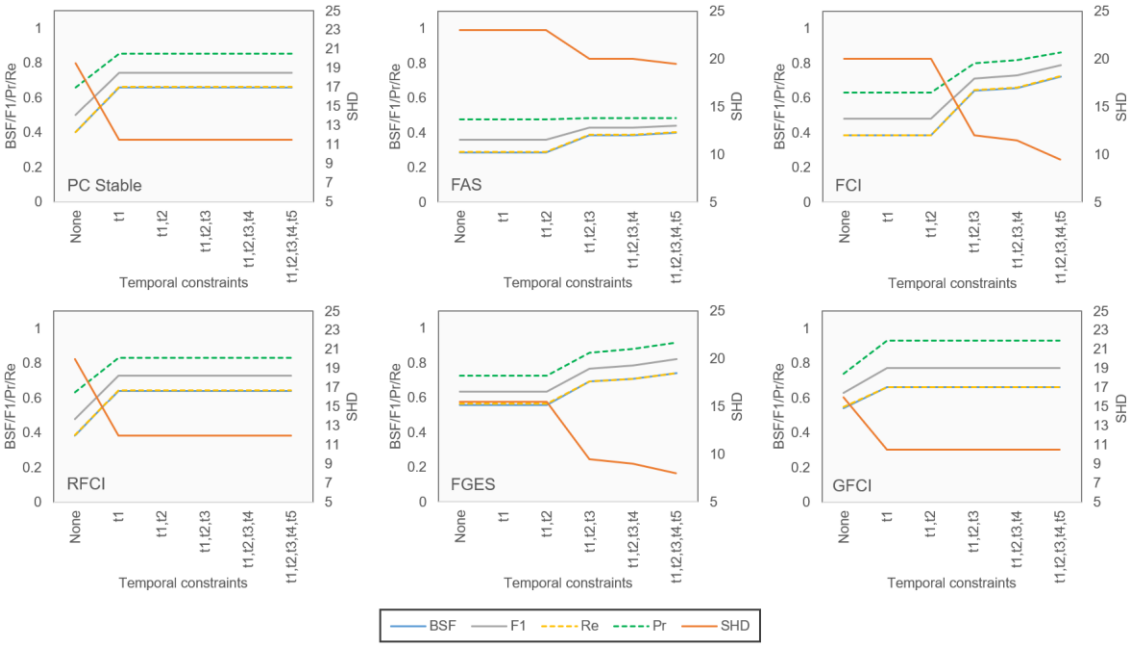


Fig 5. The accuracy scores derived from the scoring metrics Pr, Re, F1 and BSF (Primary axis) and SHD (Secondary axis), illustrating the change in accuracy over different levels of temporal constraints applied to the property market case study.

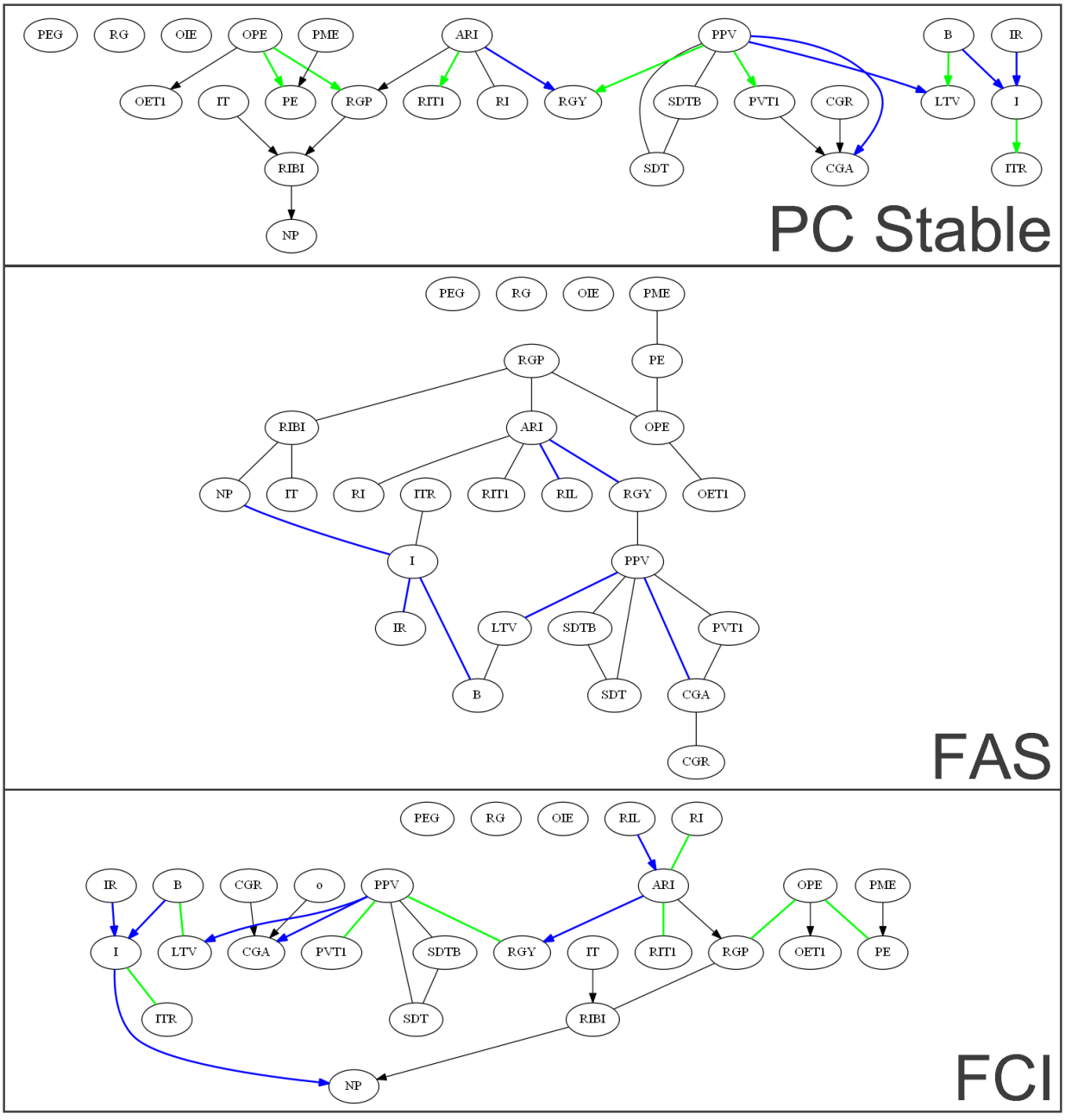


Fig 6. The graphs generated by PC-Stable, FAS, and FCI algorithms, based on the property market case study and the temporal constraints specified in Table 4. New edges resulting from the temporal constraints are shown in blue colour, reoriented edges (including undirected) are shown in green colour, and edges deleted are shown in red colour.

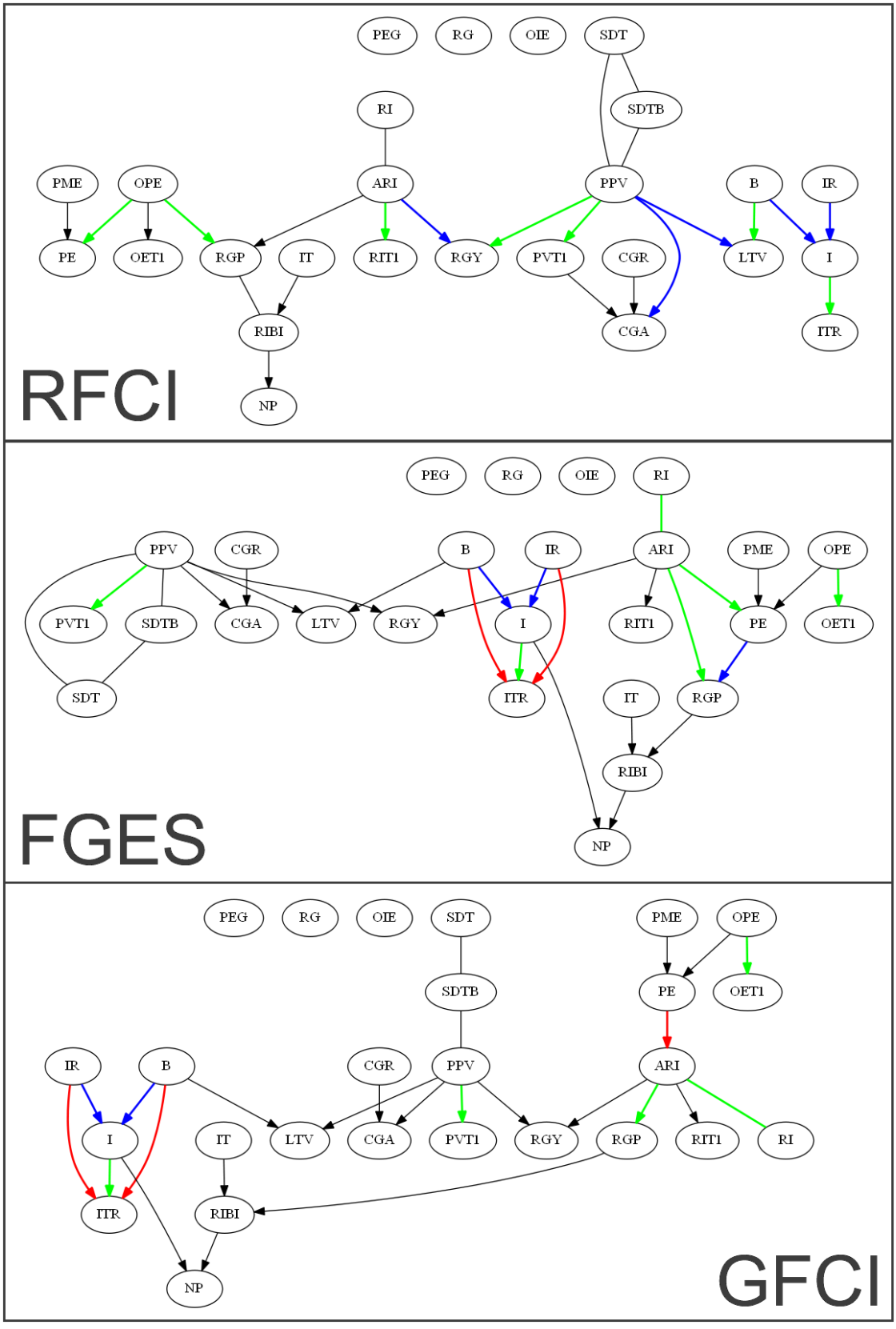


Fig 7. The graphs generated by RFCI, FGES, and GFCI algorithms, based on the property market case study and the temporal constraints specified in Table 4. New edges resulting from the temporal constraints are shown in blue colour, reoriented edges (including undirected) are shown in green colour, and edges deleted are shown in red colour.

4 DISCUSSION AND CONCLUDING REMARKS

The aim of the paper is not to demonstrate that temporal constraints are beneficial for BN structure learning. This is because it is already widely accepted that BN structure learning algorithms benefit from temporal constraints. Yet, temporal information is still largely overlooked and only some of these algorithms are designed to consider it. This is partly because temporal information is generally viewed as a form of knowledge that is subjective, rather than as a form of data that tends to be objective.

The paper focused on real-world case studies that incorporate interesting properties of transparent and objective temporal information to highlight the potential gain in accuracy that is typically lost simply because this information is not recorded as hard evidence in data. Specifically,

i. The first case study is based on a simple and clean dataset with complete objective information about the temporal order of the variables. The structure learning algorithms failed to determine the correct direction of the edges between variables prior to incorporating temporal constraints. However, some algorithms only needed a single piece of temporal information to correctly determine the true graph, while others failed to do so even after providing complete temporal information as constraints into their structure learning process. Complete temporal information has improved the scores produced by the various scoring metrics, that judge how well a learnt graph approximates the ground truth graph, by 67.1% to 79%, on average.

ii. The second case study is based on a relatively complex problem, with noisy and incomplete data as well as incomplete, although objective, information about the temporal order of some of the variables in the data. While the temporal information available for this case study amounted to just three (out of possible 56) temporal tiers with only 17 of the 56 variables assigned to one of those tiers, parts of some of the learnt graphs were still subject to major modifications given the temporal constraints. However, the results from the scoring metrics suggest that the graphical revisions have led to minor improvements (-0.5% to 7.4%, on average) relative to the improvements observed in the first case study.

iii. The third case study is based on synthetic data sampled from a real-world BN of moderate complexity. The temporal information used in this case study was also incomplete, although richer compared to that used in the second case study. The constrains involved five (out of possible 27) tiers of temporal information with all the 27 variables assigned to one of those tiers. The results from these experiments suggest that while synthetic data tends to overestimate real-world performance, incomplete temporal information still improved the scores of the learnt graphs by 43.3% to 45.6%, on average.

We often have access to objective temporal information irrespective of the application domain. However, because classical statistics and machine learning are traditionally not concerned with causal inference, the sequence of events occurring in the real world is generally reduced to an insignificant piece of information. While temporal information is clearly useful in causal inference, it is still overlooked partly because it is considered as part of knowledge-based constraints that only some of the structure learning algorithms consider.

When Cooper and Herskovits (1991) first published the K2 algorithm three decades ago, they made it dependent on knowledge about the temporal order of the variables. However, such a strong restriction represents the other extreme of the argument, as well as an inconvenience given that objective temporal information is not generally available for all the variables in the data. Moreover, because temporal information was pitched as a knowledge-based constraint, the requirement for this information understandably raised comments similar to: "*what artificial intelligence is after is the development of an agent which has some hope of overcoming problems on its own, rather than requiring engineers*" (Korb and Nicholson, 2011).

Initially, Pearl and Verma (1994) stated that "*we must still identify the clues that prompt people to perceive causal relations in the data, and we must find a computational model that emulates this perception*". However, it is now understood that the development of human knowledge is not restricted to statistical observations (Pearl and Mackenzie, 2018). Much of our causal knowledge is established by observing chains of events that enable us to experience the perception of time. If we expect machines to become rational agents in a world that requires causal perception, then we may have to provide them with something more than a dataset consisting of mere static observations under the assumption that answers about causality can be retrieved from a static observational dataset.

A possible way forward is to extend observational data in ways that capture objective temporal information for some, or all if available, the variables in the data. This will ensure that objective temporal information is viewed as part of available observational data that is generally assumed to be objective. Moreover, temporal information could be reused across similar studies without requiring access to expertise or knowledge. Lastly, the benefits of temporal constraints extend to aspects not covered in this paper, such as 'relaxing' the NP-hardness by reducing the search space of possible graphs that explain the data, as well as leading to more accurate causal models that enable the simulation of interventions for optimal decision making.

ACKNOWLEDGEMENTS

This work was supported by the ERSRC UKRI Innovation Fellowship project EP/S001646/1: *Bayesian Artificial Intelligence for Decision Making under Uncertainty* (Constantinou, 2018), and by The Alan Turing Institute in the UK under the EPSRC grant EP/N510129/1.

REFERENCES

Chickering, D. M. (2002). Optimal structure identification with greedy search. *Journal of Machine Learning Research*, vol. 3, pp. 507–554.

Colombo, D., Maathuis, M., Kalisch, M., and Richardson, T. S. (2012). Learning High-Dimensional Directed Acyclic Graphs with Latent and Selection Variables. *The Annals of Statistics*, vol. 40, Iss. 1, pp. 294–321.

Colombo, D., and Maathuis, M. H. (2014). Order-Independent Constraint-Based Causal Structure Learning. *Journal of Machine Learning Research*, vol. 15, pp. 3921–3962.

Constantinou, A., Freestone, M., W, M., Fenton, N., and Coid, J. W. (2015). Risk assessment and risk management of violent reoffending among prisoners. *Expert Systems with Applications*, vol. 42, Iss. 21, pp. 7511–7529.

Constantinou, A. C., and Fenton, N. (2017). The future of the London Buy-To-Let property market: Simulation with Temporal Bayesian Networks. *PLoS ONE*, 12(6), e0179297.

Constantinou, A. (2018). Bayesian Artificial Intelligence for Decision Making under Uncertainty. *Engineering and Physical Sciences Research Council (EPSRC)*.

Constantinou, A. (2019). Evaluating structure learning algorithms with a balanced scoring function. *arXiv 1905.12666 [cs.LG]*.

Cooper, G. F. and E. Herskovits (1991). A Bayesian method for constructing Bayesian belief networks from databases. In *Proceedings of the 7th Conference on Uncertainty in Artificial Intelligence (UAI91)*, pp. 86–94.

Cooper, G. F., and Herskovits, E. (1992). A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, vol. 9, pp. 309–347.

Dawid, A. P., Musio, M., and Stephen, F. (2015). From Statistical Evidence to Evidence of Causality. *Bayesian Analysis*, vol. 11, Iss. 3, pp. 725–752.

Fenton, N., and Neil, M. (2012). *Risk assessment and decision analysis with Bayesian networks*. London: CRC Press.

Freedman, D., and Humphreys, P. (1999). Are there algorithms that discover causal structure? *Synthese*, vol. 121, pp. 29–54.

Glymour, C., and Cooper, G. F. (1999). *Computation, Causation & Discovery*. Menlo Park, California, Cambridge Massachusetts, London, England: AAAI Press/MIT Press.

Korb, K., and Nicholson, A. (2011). *Bayesian Artificial Intelligence (Second Edition)*. London, UK: CRC Press.

Koski, T. J., and Noble, J. M. (2012). A review of Bayesian Networks and Structure Learning. *Mathematica Applicanda*, vol. 40, Iss. 1, pp. 53–103.

Meek, C. (1997). Graphical Models: Selecting causal and statistical models. PhD thesis, Carnegie Mellon University.

Ogarrio, J. M., Spirtes, P., and Ramsey, J. (2016). A Hybrid Causal Search Algorithm for Latent Variable Models. In *Proceedings of the JMLR: Workshop and Conference Proceedings*, pp. 368–379.

Pearl, J., and Verma, T. (1994). A Theory of Inferred Causation. *Logic, Methodology and Philosophy of Science*, IX, pp. 789–811.

Pearl, J., and Mackenzie, D. (2018). *The Book of Why: The New Science of Cause and Effect*. New York: Basic Books.

Schmidt, M., Niculescu-Mizil, A., and Murphy, K. (2007). Learning graphical model structure using L1-regularization paths. In *Proceedings of the 22nd National Conference on Artificial Intelligence (AAAI'07)*, pp. 1278–1283.

Spirtes, P., and Glymour, C. (1991). An Algorithm for Fast Recovery of Sparse Causal Graphs. *Social Science Computer Review*, vol. 9, Iss. 1.

Spirtes, P., Glymour, C., and Scheines, R. (2001). *Causation, Prediction, and Search: 2nd Edition*. Cambridge, Massachusetts, London, England: The MIT Press.

TETRAD. (2017). Tetrad Manual. Retrieved November 12, 2018, from http://www.phil.cmu.edu/tetrad/revised%20Tetrad%20Manual-JUNE-2017.pdf

Tsamardinos, I., Brown, L. E., and Aliferis, C. F. (2006). The Max-Min Hill-Climbing Bayesian Network Structure Learning Algorithm. *Machine Learning*, vol. 65, pp. 31–78.

Verma, T. S., and Pearl, J. (1990). Equivalence and synthesis of causal models. In *Proceedings of the 6th Annual Conference on Uncertainty in Artificial Intelligence (UAI'90)*, pp. 255–270.

Zhang, J. (2008). On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. *Artificial Intelligence*, vol. 172, Iss. 16–17, pp. 1873–1896.